블록체인 트랜잭션을 활용한 클라우드 스토리지 데이터 책임 추적성 확보 방안 연구

박병주*, 곽진**
*아주대학교 컴퓨터공학과 정보보호응용및보증연구실
**아주대학교 사이버보안학과
e-mail:bjpark.isaa@gmail.com*, security@ajou.ac.kr**

Framework for Securing Accountability of Cloud Storage Data by using Blockchain Transaction

Byeong-ju Park*, Jin Kwak**
*ISAA Lab., Department of Computer Engineering, Ajou University
**Department of Cyber Security, Ajou University

요 약

ICT 기술의 발달과 함께 클라우드의 사용이 활발해지고 있으며, 클라우드의 활용성 또한 증가하고 있다. 클라우드는 각각의 활용 용도에 따라 다양한 데이터가 저장되고 있으며, 클라우드 스토리지와 클라우드 스토리지에 저장된 데이터의 중요성 또한 증가하고 있다. 또한, 클라우드를 사용하는 이용자의 수가 증가하며 CSP에 아웃소싱 되는 데이터의 양이 증가하고 있지만, 지속적으로 보안사고가 발생하고 있으며, 신뢰 되지 않는 클라우드 환경에서는 악의적 사용자 또는 CSP에 의해 데이터 액세스 로그가 위조되거나 생략이 가능해 수정 불가능한 로깅 등을 통한 책임 추적성 확보가 필요하다. 따라서 이와 같은 문제를 해결하고 클라우드 스토리지 데이터의 책임 추적성 확보를 위해 본 논문에서는 블록체인 위·변조 불가능한 특성을 활용하여 신뢰 가능한 데이터 액세스 로깅을 통해 데이터 책임 추적성 확보가 가능한 프레임워크를 제안한다.

1. 서론

최근 ICT 기술이 발달함에 따라, 클라우드 컴퓨팅이 점점 주목받고 있다. 클라우드 스토리지는 시간이나 지역에 무관하게 접근할 수 있으며, 자신의 데이터를 다른 사용자들과 쉽게 공유할 수 있다는 장점을 보유하고 있다. 하지만 기존에 존재하던 클라우드 보안위협과 사용자가 물리적으로 데이터를 소유하는 것이 아닌, 클라우드 스토리지에 저장하고 있어 클라우드 스토리지에 저장된 사용자 데이터에 대해 책임 추적성이 요구되고 있다.

또한 클라우드 서비스를 이용하는 사용자가 증가함에 따라 클라우드 서비스를 제공하는 CSP(Cloud Service Provider)에 더 많은 데이터가 아웃소싱 되었지만, 최근 지속적으로 데이터 손실·유출 사고가 발생하고 있어 로깅 등을 통한 데이터 책임 추적성이 요구되고 있다. 일반적인 CSP는 로깅 메커니즘을 통해 로그를 생성하고 클라우드에 저장한다. 하지만 신뢰할 수 없는 클라우드 환경에서 악의적인 사용자 또는 CSP에 의해 로그는 임의적 조작이 가해질 가능성이 존재한다[1].

본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학ICT연구센터육성 지원사업의 연구결과로 수행되었음(IITP-2017-2016-0-00304)

최근 연구된 클라우드 책임 추적성 확보 방안의 경우, 신뢰 되지 않은 클라우드 환경에서 로그 무결성 보호에 초점을 두고 연구되고 있으며, 자동화된 로깅 메커니즘을 통해 로그를 생성하고 있다. 하지만, 자동 로깅 메커니즘의 경우, 분산 스토리지 시스템을 대상으로 설계되어 데이터 소유자에게 부담을 가져오며, CSP의 기능을 충분히 활용하지 못하고 있다. 또한, 신뢰할 수 없는 CSP의 경우 키 남용 공격과 같은 문제점이 발생할 수 있다. 악의적 사용자는 CSP와의 공모를 통해 레코드 생성을 우회하여 저장된 암호화된 데이터에 직접 액세스하고 유출된 키로 데이터를 복호화하는 공격도 발생 가능하다[2].

따라서, 본 논문에서는 위·변조가 불가능한 블록체인을 활용해 로깅을 수행하여 로그에 대한 신뢰성을 높이고 데이터 책임 추적성을 안전하게 제공할 수 있는 프레임워크를 제안하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서 클라우드 스토리지 데이터 책임 추적성과 기존 책임 추적성 확보 방안에 대해 연구하고, 3장에서 블록체인을 활용한 클라우드 스토리지 데이터 책임 추적성 확보가 가능한 프레임워크를 제안하고 4장에서 제안한 프레임워크에 대한 안전성을 분석하며, 마지막으로 5장에서 결론을 맺는다.

2. 관련연구

2.1 클라우드 컴퓨팅에서의 책임 추적성

클라우드 컴퓨팅 환경에서 책임 추적성은 내부 및 외부의 기준을 준수하고 적절한 조치를 취하고, 해당 조치를 설명하고 정당화하며, 문제 해결을 위한 관리 규칙을 정의한다. 클라우드에서 책임 추적성은 예비 책임(Prospective Accountability)과 소급 책임(Retrospective Accountability)으로 분류가 가능하다[3]. 최근 활발히 연구가 진행되고 있는 탐지적 제어(소급 책임)에서는 감사, 추적, 보고 및 모니터링을 포함하고 있다[4]. 클라우드 컴퓨팅 환경에서 클라우드 스토리지 데이터의 책임 추적성을 만족하기 위해서는 다음의 요소를 만족해야 한다.

❑ 증명의 신뢰성

(1) 콘텐츠 무결성 : 자격 증명을 수행하는 과정에서 사용자가 CSP에 전송하는 증명 콘텐츠는 위조가 불가능해야 한다.

(2) 증명 은폐 불가 : CSP가 자격 증명 로그를 남기는 과정에서 오작동 된 기록을 은폐할 수 없어야 한다.

(3) 로깅 회피 불가 : 악의적 사용자가 로그를 남기지 않고 클라우드 스토리지 데이터에 직접 접근해 데이터를 가져갈 수 없어야 한다.

2.2 기존 클라우드 스토리지 데이터 책임 추적성 연구

❑ 부인 방지가 가능한 로깅 절차

R. A. Popa의 연구에서는 로그의 포맷과 로깅 절차를 부인 방지가 가능하도록 정의했다[5]. 해당 로그들은 수정에 대한 저항성을 보유하고 있지만, 정당하지 않은 엔티티가 로그를 생성할 가능성도 존재한다.

❑ 자동화된 로깅 메커니즘

데이터 공유를 지원하는 분산 프레임워크를 제공하는 S. Sundareswaran의 연구는 로깅의 신뢰성을 보장하기 위해 강제 로깅 메커니즘을 수행한다. 다른 사용자가 데이터에 접근하면 자동적으로 로깅이 수행된다. 하지만 분산 프레임워크의 특성상 로그에 의해 소비되는 스토리지와 네트워크 통신량의 부하가 높아 데이터 소유자에게 높은 부담을 요구한다[6].

❑ 암호화된 데이터 스토리지

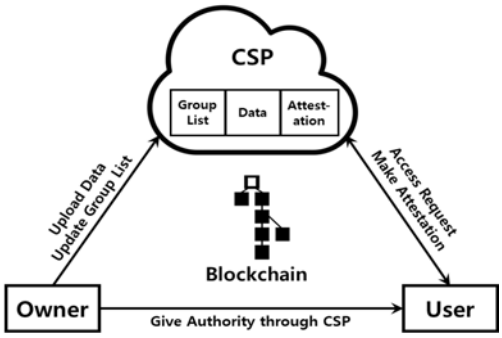
F. Xhafa 등의 연구는 속성 기반 암호화(ABE; Attribute-based Encryption)를 이용해 세분화된 접근 제어 및 책임 추적성을 제시한다[7]. 하지만 키 남용 또는 악성 키 생성기 이슈에 대응하기 위해 ABE를 사용하는 기존의 스킴들의 경우, 데이터 액세스에 많은 시간을 소비한다.

<표 1> 기존 연구의 한계점

연 구	한 계 점
R. A. Popa[5]	정당하지 않은 엔티티의 로그 생성
S. Sundareswaran[6]	데이터 소유자에게 높은 부하
F. Xhafa[7]	데이터 액세스에 과도한 시간 소비

3. 제안 사항

제안하는 프레임워크는 클라우드 스토리지 데이터의 책임 추적성을 확보하기 위한 기술로써, 사용자의 익명 ID 생성 단계, 자격 증명 및 트랜잭션 생성 단계로 이루어진다. 제안하는 프레임워크의 전체적인 아키텍처는 다음 (그림 1)과 같으며 사용되는 파라미터 값들은 <표 2>와 같다.



(그림 1) 제안 프레임워크 아키텍처

❑ Owner

데이터 소유자인 Owner는 CSP에 데이터를 업로드하고 User가 $hashID$와 $token$을 통해 CSP의 데이터에 대한 접근이 가능하도록 한다.

❑ User

사용자는 CSP를 통해 전달받은 값들을 통해 CSP에 접근을 요청하고 자격 증명을 수행한다.

❑ CSP

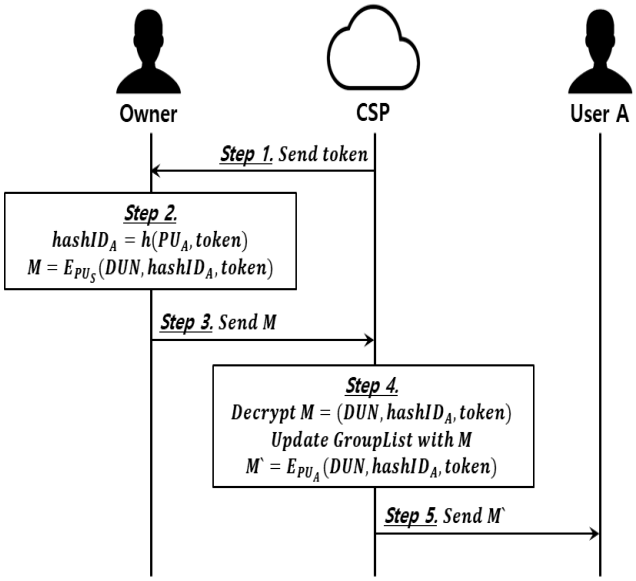
CSP는 접근 요청이 들어올 경우, Group List를 통해 권한을 확인하고 사용자의 접근을 제어한다. 인가 또는 비인가된 모든 접근에 대해 로그를 생성하고 해당 로그를 트랜잭션화해 로그에 대한 위·변조를 방지한다.

<표 2> 표기법

표기법	설 명
PR_x	x의 개인 키
PU_x	x의 공개키
$hashID_x$	x의 식별자
$h(\ )$	해시함수
DUN	데이터 고유 번호
T	트랜잭션
$token$	토큰 값

3.1 익명 ID 생성 단계

제안하는 프레임워크에서는 데이터 책임 추적성 확보를 위해 블록체인의 트랜잭션을 이용한다. 블록체인 트랜잭션에 저장되는 정보는 3.2절에서 생성되는 자격 증명을 통한 로그 데이터로 트랜잭션에 저장할 때, 별도의 암호화를 거치지 않는다. 트랜잭션에 접근 가능한 다수의 사용자가 User A의 정확한 신원을 알 수 없도록 CSP를 통해 Owner와 User A는 익명 ID를 주고받으며, 다음 (그림 2)와 같다.



(그림 2) 익명 ID 생성 단계

Step 1. CSP는 익명 ID 생성을 위해 데이터 소유자인 Owner에게 $token$을 전송한다.

Step 2. Owner는 $token$과 사용자 A의 공개키를 통해 익명 ID를 생성한다. 또한, 생성된 익명 ID와 $token$, 사용자 A에게 접근을 허용할 데이터의 고유번호를 자신의 공개키로 암호화한다.

Step 3. Owner는 Step 2.에서 생성한 정보를 CSP를 통해 사용자 A에게 전달한다.

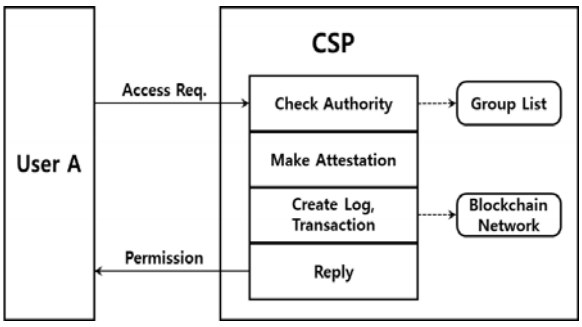
본 절의 익명 ID 생성 단계와 동일한 절차를 거쳐, Owner는 익명 ID가 저장된 CSP의 Group List에 추가 및 갱신할 수 있다. Group List는 다른 사용자들의 접근이 불가능하도록 암호화되어 저장한다. Group List는 다음 (그림 3)과 같은 구조를 갖는다.

hashID	token	접근 가능한 DUN	Nonce

(그림 3) Group List 구조

3.2 자격 증명 및 트랜잭션 생성 단계

자격 증명 및 트랜잭션 생성 단계에서는 CSP가 사용자에 대한 자격 증명을 수행하고 검증을 통과하면 로그를 생성한다. 또한, 로그를 포함하는 블록체인 트랜잭션 생성을 통해 로그에 대해 위·변조를 방지하고 지속적인 데이터 책임 추적성 보장이 가능하도록 한다. 자격 증명 및 트랜잭션을 생성하는 프로토콜은 다음 (그림 4)와 같다.



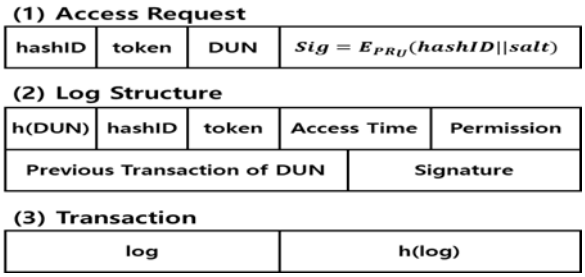
(그림 4) 자격 증명 및 트랜잭션 생성 단계

Step 1. 사용자 A는 CSP에 데이터에 대해 접근을 요청한다.

Step 2. CSP는 Group List를 통해 사용자 A의 권한 보유 여부를 조회한다.

Step 3. 자격 증명이 이루어지고 사용자 A의 데이터 접근에 대한 로그가 생성되고 이를 통해 트랜잭션이 생성되어 블록체인 네트워크에 브로드캐스트 된다.

Step 4. 데이터에 대한 접근 권한 보유 여부에 따라, CSP는 사용자 A에게 승인/거부의 응답 메시지를 전송한다.



(그림 5) 자격 증명 및 로깅에 사용되는 데이터 구조

사용자가 CSP에 자신의 권한을 증명하기 위해 3.1절에서 CSP로부터 전송받은 데이터를 통해 (그림 5)의 (1)을 자신의 개인 키로 암호화한 메시지를 통해 자격 증명을 요청한다. CSP에서 Group List를 통해 자격 증명을 수행한 후, 승인 또는 거부된 데이터에 대한 모든 접근에 대해 위의 (그림 5)의 (2)과 같은 구조의 로그를 생성한다. 또한, 해당 로그를 기반으로 (그림 5)의 (3)와 같은 정보를 포함하는 트랜잭션을 생성하여 데이터 접근 로그를 블록체인 네트워크에 브로드캐스트한다.

4. 안전성 분석

본 장에서는 2.1 절에서 분석한 책임 추적성을 확보하기 위한 요구사항과 블록체인을 도입하면서 발생할 수 있는 다양한 보안 문제에 대해 기존 연구와의 비교를 통해 안전성 분석을 수행한다.

❑ 증명의 신뢰성 보장

(1) 콘텐츠 무결성 : 자격 증명을 수행하기 위해 사용자가 CSP에 전송하는 자신의 증명 콘텐츠는 (그림 5)의 (1)과 같은 구조를 갖는다. 해당 구조에서는 사용자의 서명을 통해 콘텐츠에 대한 위·변조를 검출할 수 있다.

(2) 증명 은폐 불가 : 제안하는 프레임워크에서 CSP는 블록체인을 통해 (그림 5)의 (2)와 같은 구조로 로그를 트랜잭션화해 저장한다. CSP의 의도적인 로그 삭제가 있을 경우, 새로운 로그를 생성하는 과정에서 이전 접근에 대한 트랜잭션 값을 얻을 수 없어 오류가 발생해 증명 은폐를 방지할 수 있다.

(3) 로깅 회피 불가 : 클라우드 스토리지의 데이터에 접근하기 위해서는 CSP가 생성한 *token*과 Owner가 생성한 *hashID*가 있어야 한다. 하지만 악의적 공격자는 해당 값들을 보유하고 있지 않으므로 데이터에 접근하는 과정에서 거부당하게 된다.

❑ 기존 연구와의 비교·분석

(1) 정당하지 않은 엔티티의 로그 생성 방지

R. A. Popa의 연구에서는 정당하지 않은 엔티티가 로그를 생성할 가능성이 존재했다. 하지만 제안하는 프레임워크에서는 CSP만이 서명을 통해 정당한 로그를 생성할 수 있다[5].

(2) 데이터 소유자의 자원 부담 절감

S. Sundareswaran의 연구 등에서 분산 프레임워크로 인해 로그 생성에 있어 데이터 소유자에게 높은 부담을 요구했지만 제안하는 프레임워크의 경우, CSP에서 로그를 생성하며 데이터 소유자는 사용자의 인증 정보만을 생성하는 연산을 수행해 데이터 소유자의 부담을 경감할 수 있다[6].

(3) 데이터 액세스 소요 시간 절감

F. Xhafa 등의 연구 등에서는 암호화된 데이터 스토리지로 높은 데이터 액세스 시간을 필요로 한다[7]. 하지만 제안하는 프레임워크의 경우, 자격 증명 절차 후 사용자의 즉각적인 접근을 허용해 데이터 액세스에 소비되는 시간을 경감할 수 있다.

❑ 트랜잭션의 공개된 로그로 인한 이슈

트랜잭션에 공개되는 로그에 포함되는 정보들은 데이터 고유번호, 익명의 *hashID*, *token*, 접근 시간, CSP의 서명 등이 있다. 하지만 공격자가 *hashID*, *token*, *DUN*을 수집하더라도 사용자의 개인 키로 생성되는 전자서명 등은 생성이 불가능해 로그 정보를 이용한 재전송 공격 등은 수행이 불가능하다.

5. 결론

클라우드의 사용이 증가하며, 클라우드 스토리지에 저장되는 데이터의 양도 크게 증가하고 있다. 신뢰 되지 않는 클라우드 환경의 경우, 다양한 형태의 공격이 가능해 이를 방지하기 위한 데이터 책임 추적성에 대한 연구가 활발하다. 하지만 기존 책임 추적성에 대한 연구들의 경우, CSP의 기능을 충분히 활용하지 못하거나, 데이터 소유자에게 부담을 가져오는 문제점을 가지고 있었다. 따라서, 본 논문에서는 위·변조가 불가능한 블록체인을 활용해 로깅을 수행하여 로그에 대한 신뢰성을 확보하고 데이터 책임 추적성을 확보하기 위한 프레임워크를 제안하였으며, 이에 대한 안전성 분석을 수행하였다.

추후 연구로는 구현을 통한 연구 결과 실증을 진행하고자 한다.

참고문헌

[1] Faheem Zafar, Abid Khan, Saif Ur Rehman Malik, Mansoor Ahmed, Adeel Anjum, Majid Iqbal Khan, Nadeem Javed, Massom Alam, Fuzel Jamil, “A Survey of Cloud Computing Data Integrity Schemes: Design Challenges, Taxonomy and Future Trends,” Computers & Security Volume 65, p29 - p49. March, 2017.

[2] S. Sundareswaran, A. Squicciarini, and D. Lin, “Ensuring distributed accountability for data sharing in the cloud,” IEEE Transactions on Dependable and Secure Computing, vol. 9, no. 4, pp. 556 - 568, 2012.

[3] Dipen Contractor, Dhiren Patel, “Accountability in Cloud Computing by Means of Chain of Trust,” International Journal of Network Security, Vol.19, No.2, PP.251-259, Mar. 2017.

[4] D. Nunez, C. Fernandez-Gago, S. Pearson, and M. Felici, “A Metamodel for Measuring Accountability Attributes in the Cloud,” in IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom), vol. 1, pp. 355 - 362, 2013.

[5] R. A. Popa, J. R. Lorch, D. Molnar, H. J. Wang, and L. Zhuang, “Enabling security in cloud storage slas with cloudproof.” in USENIX Annual Technical Conference, vol. 242, 2011.

[6] S. Sundareswaran, A. Squicciarini, D. Lin, and S. Huang, “Promoting distributed accountability in the cloud,” in Cloud Computing (CLOUD), 2011 IEEE International Conference on. IEEE, 2011, pp. 113 - 120.

[7] F. Xhafa, J. Feng, Y. Zhang, X. Chen, and J. Li, “Privacy-aware attributebased phr sharing with user accountability in cloud computing,” The Journal of Supercomputing, vol. 71, no. 5, pp. 1607 - 1619, 2015.